

# 모두의핏 DATA FLOW

## 1. 문서 개요

### 1.1 문서명

모두의핏 모바일 앱 - Data Flow 문서

### 1.2 목적

본 문서는 모두의핏 사용자 앱의 주요 기능 수행 시  
데이터가 생성 · 조회 · 수정 · 삭제되는 전체 과정을 명확히 정의하여  
백엔드 · 프론트 · DB 설계 및 QA 테스트 기준을 제시하는 것을 목적으로 한다.

### 1.3 범위

사용자 앱(홈, 탐색, 상세, 예약, 커뮤니티, 마이페이지)의 데이터 흐름

API 기반 Data Flow

주요 엔티티 단위 Data Flow

CRUD 관점 데이터 작동 방식

운영자/지도자/관리자 영역은 별도 문서로 분리

## 2. 전체 데이터 흐름 요약

사용자 행동(Input)



App UI(이벤트 발생)



서버 API Request



Backend Logic



Database Read/Write

↓

서버 Response(JSON)

↓

App UI Rendering

즉, 모든 기능은  
사용자 행동 → API 호출 → DB 처리 → 응답 → 화면 표시  
의 사이클로 처리된다.

### 3. 주요 데이터 엔티티 구조

User(사용자)

Facility(시설)

Leader(지도자)

Program(프로그램/강습)

Review(후기)

Reservation(예약)

Community(게시판/댓글)

Notice(공지)

각 기능의 Data Flow는 이 엔티티를 중심으로 이루어진다.

### 4. 기능별 상세 Data Flow

아래는 실제 개발 단계에서 API, DB, 화면이 어떻게 데이터를 교환하는지 정의한 흐름이다.

#### 4.1 홈(Home) Data Flow

① 내 주변 시설 추천

GPS 좌표 획득(App)

↓

/facility/nearby API 호출

↓  
Backend: 거리 기반 Facility 검색

↓  
DB: 시설 정보(Facility Table) 조회  
↓  
응답(JSON)

↓  
App: 추천 카드 리스트 렌더링  
② 내 주변 지도자 추천  
/leader/nearby API  
→ Leader Table 조회  
→ 거리 계산  
→ 응답  
→ UI 표시  
③ 공지/배너  
App → /notice/list

↓  
Notice Table 조회  
↓  
응답 → UI 표시

#### 4.2 탐색(Search) Data Flow

① 시설 검색(Search Facility)

검색어 입력(App)

↓  
/facility/search?q=  
↓  
DB: 시설명 LIKE 검색  
↓  
결과 리스트 응답

↓  
App: 리스트 표시  
② 필터 검색(Search Filter)  
필터 값 선택(App)

↓  
/facility/filter API  
↓  
Backend: 조건 Query 생성

↓  
DB: 조건 검색  
↓  
결과 응답  
③ 지도자 검색(Search Leader)  
검색어/필터 입력

↓  
/leader/search  
↓  
DB: 종목/경력/지역 조건 조회  
↓  
결과 응답 → UI 표시

#### 4.3 시설 상세(Facility Detail) Data Flow

① 시설 기본 정보 조회

App → /facility/detail?id=  
↓  
DB: Facility, Program, ReviewCount JOIN 조회  
↓  
JSON 응답  
↓  
App: 탭(정보/후기/프로그램) 렌더링

② 프로그램 목록 조회

/program/list?facilityId=

→ DB 조회 → 응답 → UI 표시

③ 후기 목록

/review/facility?id=

→ Review Table 조회

→ 응답 → UI 표시

#### 4.4 지도자 상세(Leader Detail) Data Flow

App → /leader/detail?id=

↓

DB: Leader 정보 & 후기 JOIN 조회

↓

응답 → App 표시

#### 4.5 예약 플로우(Reservation Flow)

가장 핵심 기능 — 실제 CRUD 발생

① 예약 가능 날짜 조회

App → /reservation/available-date?facilityId=

↓

DB: Reservation Table 조회

↓

Backend: 날짜별 예약 가능 여부 계산

↓

가능 날짜 배열 응답

↓

App 달력 표시

② 예약 가능 시간 조회

App(날짜 선택)

↓

/reservation/available-time



DB: 해당 날짜 예약 상태 조회



가능/불가 시간슬롯 계산



응답 → 시간선택 UI 표시

③ 예약 생성(Create Reservation)

App: (시설ID/지도자ID + 날짜 + 시간) 전송



/reservation/create



Backend: Validation(중복체크)



DB Insert(Reservation Table)



예약 완료 응답(JSON)



App: 예약완료 화면 표시

④ 예약 내역 조회(My Reservation List)

App → /reservation/list?userId=



DB: Reservation Table에서 특정 User 예약 조회



리스트 응답

⑤ 예약 취소(Cancel)

App → /reservation/cancel?id=



DB: status = 'cancel' 업데이트



응답 → UI 갱신

#### 4.6 커뮤니티(Community) Data Flow

① 게시글 목록 조회

/community/list

→ DB: Community Table 조회

→ 응답 → UI 렌더링

② 게시글 상세 조회

/community/detail?id=

→ 게시글 + 댓글 조회

→ UI 표시

③ 댓글 작성

App 입력 → /comment/create

↓

DB Insert(Comment Table)

↓

응답 → 화면 즉시 갱신

#### 4.7 마이페이지(My Page) Data Flow

① 내 후기 목록 조회

/review/my?userId=

→ Review Table 조회

→ 응답

② 사용자 정보 조회

/user/info

→ User Table 조회

→ 응답

#### 5. Data Flow 시퀀스 요약 (전체)

1) 사용자 입력

2) 화면(UI)에서 이벤트 발생

- 3) API 호출
- 4) 서버 로직 수행
- 5) DB 읽기/쓰기
- 6) 응답(JSON)
- 7) UI 업데이트